

SPECIFICATION AMENDMENTS

None

CLAIM AMENDMENTS

Claim Amendment Summary

Claims pending

- Before this Amendment: Claims 1-31 and 39-54.
- After this Amendment: Claims 1, 3-7, 10-11, 14-20, 22, 24-27, 29-31, and 39-54.

Non-Elected, Canceled, or Withdrawn claims: 2, 8-9, 12-13, 21, 23, and 28.

Amended claims: Claims 1, 3-7, 10-11, 14-20, 22, 24-27, 29-31, 39-41, 48-49, and 51.

New claims: None.

Claims:

1. (Currently Amended) One or more [computer-readable] computer-storage media comprising a plurality of data structures, the plurality of data structures comprising:

a code unit having executable instructions;

one or more rewriters, wherein each rewriter modifies the code unit in a unique way; and

a rewriter list identifying at least one rewriter, wherein the rewrite list identifies which rewriter to apply to rewrite the code unit and provides the sequence the rewriters are applied to the code unit; and

~~[one or more rewriters that include the at least one rewriter, each rewriter capable of rewriting the code unit]~~

a rewrite manager having executable instructions configured to access one or more rewriters and execute the one or more rewriters against the code unit to generate a rewritten code unit, whereby the rewrite manger identifies and sequences the appropriate rewriters by consulting the rewrite list.

2. (Canceled)

3. (Currently Amended) One or more ~~[computer-readable]~~ computer-storage media as recited in claim 1, further comprising a cache, the rewrite manager further configured to store the rewritten code unit in the cache.

4. (Currently Amended) One or more ~~[computer-readable]~~ computer-storage media as recited in claim 1, wherein the rewriter list is selected from ~~[the]~~ a group comprising:

- a list of one or more custom attributes within the code unit;
- a list within a security policy;
- a list within an installation tool;

a list within a configuration file; and
a list within an XML (Extensible Markup Language) file.

5. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 2, wherein the rewrite manager is a module selected from [~~the~~] a group comprising:

- a stand-alone module;
- an operating system module;
- an execution environment module;
- a JIT compiler module;
- a source code compiler module; and
- an installation tool configured to install the code unit.

6. (Currently Amended) A computer comprising the one or more [~~computer-readable~~] computer-storage media recited in claim 1, the computer selected from [~~the~~] a group comprising:

- a developer computer configured to create the code unit;
- an intermediate computer configured to deploy the code unit; and
- a deployment computer configured to execute the code unit.

7. (Currently Amended) One or more [~~computer-readable~~] computer-storage media comprising a code unit, the code unit including executable instructions configured for:

- initiating a transformation of the code unit; and

identifying and sequencing one or more rewriters to implement the transformation, each rewriter configured to implement a unique transformation of the code unit, wherein the one or more rewriter are identified and sequenced by a rewrite manager having executable instructions configured to access a rewrite list, wherein the rewrite list identifies which rewriter to apply to rewrite the code unit and provides the sequence the rewriters are applied to the code unit.

8. (Canceled)

9. (Canceled)

10. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 7, wherein the initiating comprises:

identifying an environment in which the transformation should be implemented; and

initiating the transformation only if the code unit experiences the identified environment.

11. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 10, wherein the identifying an environment comprises:

identifying a source code compilation environment;

identifying a pre-execution-compilation environment;

identifying a compilation-on-installation environment;
identifying an execution environment; and
identifying an installation environment.

12. (Canceled)

13. (Canceled)

14. (Currently Amended) One or more [~~computer-readable~~]
computer-storage media comprising computer-executable instructions configured
for:

receiving a code unit having executable instructions;
determining at least one rewriter with which the code unit may be
rewritten, wherein each rewriter modifies the code unit in a unique way;

calling the at least one rewriter, wherein the rewriter is called by a rewrite
manager having executable instructions configured to access one or more
rewriters by accessing a rewrite list that identifies which rewriter to apply to
rewrite the code unit and provides the sequence the rewriters are applied to the
code unit; and

executing the at least one rewriter against the code unit to generate a
rewritten code unit.

15. (Currently Amended) One or more [~~computer-readable~~]
computer-storage media as recited in claim 14, comprising further computer-

executable instructions configured for verifying trustworthiness of the code unit and the at least one rewriter prior to the executing.

16. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 15, wherein the verifying comprises authenticating a digital signature.

17. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 14, comprising further computer-executable instructions configured for storing the rewritten code unit in a cache.

18. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 17, comprising further computer-executable instructions configured for:

- receiving an instruction to execute the code unit;
- recognizing that the code unit has been rewritten;
- loading the rewritten code unit from the cache; and
- executing the rewritten code unit.

19. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 17, comprising further computer-executable instructions configured for:

- generating a digital signature for the rewritten code unit; and
- associating the digital signature with the rewritten code unit.

20. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 14, wherein the at least one rewriter is a plurality of rewriters, the one or more computer-readable media comprising further computer-executable instructions configured for sequencing the plurality of rewriters to rewrite the code unit in a particular rewrite order.

21. (Canceled)

22. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 21, wherein the accessing comprises accessing the rewriter list in a location selected from [~~the~~] a group comprising:

the code unit;

a separate file associated with the code unit; and

a system policy.

23. (Canceled)

24. (Currently Amended) One or more [~~computer-readable~~] computer-storage media as recited in claim 14 embodied in a tool selected from [~~the~~] a group comprising:

a source code compiler;

an installation tool;

a managed execution environment;

a stand-alone rewrite management tool; and
a JIT (just in time) compiler.

25. (Currently Amended) A computer device comprising the one or more [~~computer-readable~~] computer-storage media recited in claim 14, the computer selected from [the] a group comprising:

a developer computer configured to develop the code unit;
an intermediate computer configured to install the code unit; and
a deployment computer configured to execute the code unit.

26. (Currently Amended) A computer apparatus comprising a computing device and memory:

a code unit;
a composable set of rewriters, each rewriter configured to rewrite the code unit in a unique manner;
a rewrite manager configured to identify one or more rewriters from the composable set of rewriters, sequence the rewriters and to execute the identified one or more rewriters against the code unit, wherein the rewrite manager determines the identity and sequence by accessing a rewrite list that identifies which rewriter to apply to rewrite the code unit and provides the sequence the rewriters are applied to the code unit; and
a rewritten code unit generated by executing the identified one or more rewriters against the code unit.

27. (Currently Amended) A computer apparatus as recited in claim 26, further comprising a rewrite cache, the rewrite manager further configured to store the rewritten code unit in the rewrite cache.

28. (Canceled)

29. (Currently Amended) A computer apparatus as recited in claim 28, wherein the rewriter list is a component selected from ~~[the]~~ a group comprising:

- a list of rewriters in the code unit;
- a list of rewriters in a stand-alone file;
- a list of rewriters in a security policy; and
- a list of rewriters in an installation tool.

30. (Currently Amended) A computer apparatus as recited in claim 26, further comprising:

- a first digital signature associated with the code unit; and
- a set of second digital signatures, each second digital signature associated with a specific rewriter from the composable set of rewriters;

wherein the rewrite manager is further configured to determine if the code unit is trusted based on the first digital signature, determine if each rewriter from the identified one or more rewriters is trusted based on a corresponding second digital signature from the set of second digital signatures, and execute the

identified one or more rewriters against the code unit only if both the code unit and each rewriter from the identified one or more rewriters are trusted.

31. (Currently Amended) A computer apparatus as recited in claim 30, further comprising a third digital signature associated by the rewrite manager with the rewritten code unit and configured to verify that the rewritten code unit is trusted.

39. (Currently Amended) A computer apparatus as recited in claim 26, wherein the rewrite manager is a component selected from [the] a group comprising:

- a stand-alone rewrite module;
- a rewrite module configured as part of an operating system;
- a rewrite module configured as part of an installation tool; and
- a rewrite module configured as part of a security policy.

40. (Currently Amended) A computer apparatus as recited in claim 26, selected from [the] a group comprising:

- a development computer configured to develop the code unit;
- an intermediate computer configured to install the code unit; and
- a deployment computer configured to execute the code unit.

41. (Currently Amended) A method comprising:
receiving an executable code unit;

determining that the code unit needs to be rewritten;

determining one or more rewriters to rewrite the code unit, wherein the rewriter is determined by a rewrite manager that accesses a rewrite list that identifies which rewriter to apply to rewrite the code unit and provides the sequence the rewriters are applied to the code unit; and

running the one or more rewriters against the code unit to generate a rewritten code unit.

42. (Original) A method as recited in claim 41, further comprising verifying trustworthiness of the code unit and the one or more rewriters prior to the running.

43. (Original) A method as recited in claim 42, wherein the verifying comprises authenticating a digital signature.

44. (Original) A method as recited in claim 41, further comprising storing the rewritten code unit in a cache.

45. (Original) A method as recited in claim 44, further comprising:
receiving an instruction to execute the code unit;
recognizing that the rewritten code unit is stored in the cache;
loading the rewritten code unit from the cache; and
executing the rewritten code unit.

46. (Original) A method as recited in claim 44, further comprising:
generating a digital signature for the rewritten code unit; and
associating the digital signature with the rewritten code unit.

47. (Original) A method as recited in claim 41, further comprising
sequencing the one or more rewriters to rewrite the code unit in a particular
rewrite order.

48. (Currently Amended) A method as recited in claim 47, wherein
the sequencing comprises:

accessing [a] the rewriter list; and
setting the rewrite order according to the rewriter list.

49. (Currently Amended) A method as recited in claim 41, wherein
the determining one or more rewriters comprises reading [a] the rewriter list.

50. (Original) A method as recited in claim 49, wherein the reading
comprises reading the rewriter list in a location selected from the group
comprising:

the executable code unit; and
a separate file associated with the executable code unit.

51. (Currently Amended) A method comprising:
receiving a code unit;

determining that the code unit is to be rewritten by a rewriter, wherein the rewriter is determined by a rewrite manager that accesses a rewrite list that identifies which rewriter to apply to rewrite the code unit and provides the sequence the rewriters are applied to the code unit;

determining if the code unit and the rewriter are trusted;

running the rewriter against the code unit to generate a rewritten code unit if the code unit and the rewriter are trusted;

storing the rewritten code unit in a cache.

52. (Original) A method as recited in claim 51, further comprising:
generating a digital signature for the rewritten code unit; and
attaching the digital signature to the rewritten code unit.

53. (Original) A method as recited in claim 52, further comprising:
receiving a call to execute the code unit;
recognizing that the code unit has been rewritten;
loading the rewritten code unit from the cache;
verifying the digital signature attached to the rewritten code unit;

and

executing the rewritten code unit if the verifying indicates that the rewritten code unit is secure.

54. (Original) A method as recited in claim 51, wherein the rewriter is an application compatibility rewriter and the determining that the code unit is to be rewritten comprises:

identifying the code unit as an application; and

consulting an application compatibility rewrite database to determine if any part of the application needs to be rewritten for compatibility with a current execution environment.